資料探勘研究與實務期末專題----

流浪貓狗在收容所最終安置之預測

第十三組

組員:

陳祈勳(0853425)

羅紹華(0853423)

范姜鈞(0853422)

陳佳祥(0853433)

一、 研究動機與目的

寵物是我們的好朋友，從以前到現在一直扮演著重要的角色，成為陪伴人生的最佳夥伴。但意外總會發生，寵物可能在出門時走失或是被人惡意棄養的事件層出不窮，令人心痛。這幾年開始，動物權的意識逐漸被重視，動物保護團體也不斷成立，而國家也立法通過「動物保護法」保護動物們，並衍生出許多相關法律，但卻無法阻止流浪動物的不斷出現。在台灣，民眾在遇到流浪動物時採取的行為基本上都是通報動保團體或是動保處處理，而這些動物的下場基本上都是送往公立收容所。但公立收容所的空間不足，動物們的生活環境低落，都會造成問題，又近年為了人道考量，不得將動物安樂死的觀念興起，導致政府也立法控管這些動物的處置方式，在處置方式越來越少的狀況，這些動物的安置成了一個大問題。

由上述的原因我們興起對流浪動物的最終安置有了興趣，於是選擇 Kaggle 平台上的「Shelter Animal Outcomes」分析競賽的題目，希望透過對美國收容所的動物進行最終狀態的結果預測，包括被領養、死亡、安樂死、被原主人領回、轉移到其他收容所。藉以讓台灣也可以有相關的照護措施，因為資源有限，在有限的條件下怎麼樣才能做到最好的照顧可以是最後的目標。

二、 資料集敘述

數據來自 Austin 動物中心從 2013 年 10 月 1 日至 2016 年 3 月間所蒐集的資料。共有 38185 筆資料，10 個變數，其中訓練集共有 26729 筆，測試集共有 11456 筆，結果代表動物離開動物中心時的所受到的處置，即為我們所要預測的目標。

表一: 各變數所代表意義

變數	意義
ANIMALID	動物的 ID 編號（動物在進入動物中心時會被賦予的編號）
NAME	動物的名稱(應為主人所取的名稱)
DATETIME	進入收容所的日期(例如: 2018/06/14 14:00)
OUTCOMETYPE	動物的最終處置（被領養、死亡、安樂死、被原主人領回、轉移到其他收容所共 5 種狀況）
OUTCOMESUBTYPE	是對動物最終狀態的補充說明(例如: Foster 為中途之家領養)
ANIMALTYPE	動物的類型（狗或貓共 2 種狀況）
SEXUPONOUTCOME	動物性別和生育能力（雄性動物不能生育、雌性動物不能生育、雄性動物能夠生育、雌性動物能生育、未知共 5 種狀況）
AGEUPONOUTCOME	動物的年齡(以年、週或月來記數)
BREED	動物的品種(例如: 美國短毛貓等 1678 種品種)
COLOR	動物的毛色(例如: 咖啡色等 411 種顏色)

三、 **分析工具**

- 開發工具:Spyder
- 開發環境:Python3.7
- 使用技術:Keras、Pandas、Sklearn 套件

四、 實作與評估方法

1.Random Forest

隨機森林是一個用隨機方式建立的，包含多個決策樹的分類器。
其輸出的類別是由各個樹輸出的類別的眾數而定。而隨機性主要體現在兩個方面：

（1）訓練每棵樹時，從全部訓練樣本（樣本數為 N）中選取一個可能有重複的大小同樣為 N 的數據集進行訓練（即 bootstrap 取樣）；

（2）在每個節點，隨機選取所有特徵的一個子集，用來計算最佳分割 方式。

其優點為：

- 對於很多種資料，可以產生高準確度的分類器
- 可以處理大量的輸入變數
- 可以在決定類別時，評估變數的重要性
- 對於不平衡的分類資料集來說，可以平衡誤差
- 訓練速度快，容易做成並行化方法
- 在訓練過程中，能夠檢測到 feature 間的 interactions
- 如果有很大一部分的特徵遺失，仍可以維持準確度

其缺點為：

- 隨機森林已經被證明在某些 noise 較大的分類或回歸問題上會 overfitting
- 對於有不同取值的屬性的數據，取值劃分較多的屬性會對隨機森林產生更大的影響，所以隨機森林在這種數據上產出的屬性權值是不可信的。

預測結果:

	precision	recall	f1-score	support
Adoption	0.63	0.87	0.73	2182
Died	0.00	0.00	0.00	47
Euthanasia	0.00	0.00	0.00	311
Return_to_owner	0.48	0.38	0.42	945
Transfer	0.74	0.63	0.68	1861
accuracy			0.64	5346
macro avg	0.37	0.38	0.37	5346
weighted avg	0.60	0.64	0.61	5346

2. Decision Tree

決策樹是一個預測模型，他代表的是對象屬性與對象值之間的一種映射關係。樹中每個節點表示某個對象，而每個分叉路徑則代表某個可能的屬性值，而每個葉節點則對應從根節點到該葉節點所經歷的路徑所表示的對象的值。決策樹僅有單一輸出，若欲有複數輸出，可以建立獨立的決策樹以處理不同輸出。

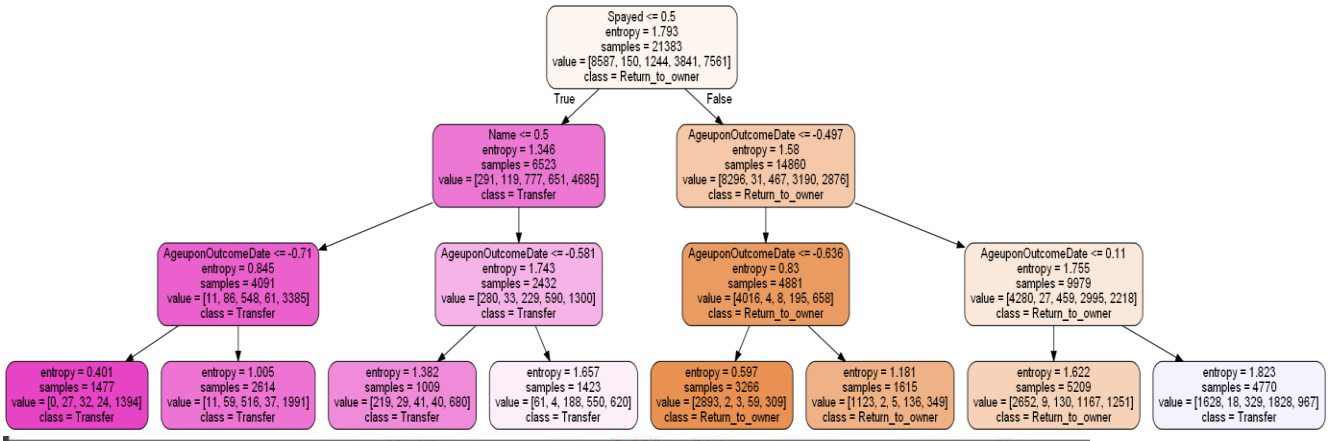
其優點為：

- 計算簡單，易於理解，可解釋性強
- 比較適合處理有缺失屬性的樣本
- 能夠處理不相關的特徵
- 在相對短的時間內能夠對大型數據源做出可行且效果良好的結果

其缺點為：

- 容易發生 overfitting（隨機森林可以很大程度上減少 overfitting）
- 忽略了數據之間的相關性
- 對於那些各類別樣本數量不一致的數據，在決策樹當中,信息增益的結果偏向於那些具有更多數值的特徵

預測結果:



	precision	recall	f1-score	support
Adoption	0.67	0.77	0.71	2182
Died	0.00	0.00	0.00	47
Euthanasia	0.54	0.06	0.11	311
Return_to_owner	0.38	0.48	0.42	945
Transfer	0.73	0.61	0.66	1861
accuracy			0.62	5346
macro avg	0.46	0.39	0.38	5346
weighted avg	0.62	0.62	0.60	5346

五、 流程

1.資料前處理

因為一開始的變數過於繁瑣且牽涉到太多的層面，所以我們將資料做以下的處理已得到新的資料集:

1. Name(動物名字):

   我們將此變數變成一二元變數，分為有無名字兩種，有名字為 1，無名字為 0。

2. DateTime(進入收容所時間):

   我們將此變數拆解開來，變成四個類別變數:

   (1)Year: 動物進收容所之年份，共有 2013、2014、2015、2016 年 4 種類別。

   (2)Month: 動物進收容所之月份，共有 1~12 月 12 種類別。

3. SexuponOutcome(性別及生育能力):

   我們利用將此變數拆成兩個類別變數:

   (1)Gender： 動物的性別，分成公、母及未知 3 個類別。

   (2)Fertility： 有無生育能力，分成有、無及未知 3 個類別。

4. AgeuponOutcome(動物年齡):

   因為此變數是以年、週、天來計數，我們將之轉為天數，並做正規化方便訓練。

5. Color(動物毛髮顏色):

   原本顏色種類有 366 種類別,我們將個數小於 100 個的種類合併成”other”，最後剩下 43 個類別。

6. AnimalType(動物種類)、Breed(品種):

   我們發現 Breed 中的資料大部分分為兩種，若品種含有”000 Mix”及”XXX / YYY”，舉例來說我們認為” Pit Bull Mix”特徵可能比較接近” Pit Bull”，所以將之分為” Pit Bull”。而”XXX / YYY”的部分會比較像前面”XXX”，如 Lhasa Apso/Miniature Poodle 稱為 Lhasa Apso。

   並將個數小於 200 的種類，依照狗跟貓分別分為”dog_others”, ”

cat_others”。 Domestic Shorthair 的個數遠大於其他種類，我們認為這會影響後續分析，所以將不同顏色的 Domestic Shorthair，視為不同種類的貓，其中個數小於 199 的分為” Others Domestic Shorthair”，最終剩下 44 個類別。

變數	意義
ANIMALTYPE	動物的類型（狗或貓，2 個類別）
NAME	動物是否有名字(類別型變數，2 個類別)
YEAR	進入收容所的年分(類別型變數，4 個類別)
MONTH	進入收容所的月份(類別型變數，12 個類別)
GENDER	動物的性別(類別型變數，3 個類別)
FERTILITY	有無生育能力(類別型變數，3 個類別)
AGES	動物的年齡(連續型變數)
COLOR	動物毛髮顏色(類別型變數，43 個類別)
BREED	動物的品種(類別型變數，49 個類別)

表二: 資料前處理完各變數及所代表意義

2.資料切割

將資料切割為 train data 和 test data

```
# # Split Data
from sklearn.model_selection import train_test_split

label = train_df['OutcomeType']
train_df = train_df.drop(columns='OutcomeType')
X_train, X_test, y_train, y_test = train_test_split(train_df, label, test_size=0.2, random_state=0)
```

3.建立模型與設定參數

```
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=2000,oob_score = True,n_jobs = -1,max_features=0.2,min_samples_leaf=50)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

```
232 #決策樹
233 from sklearn import tree
234 clf = tree.DecisionTreeClassifier(criterion='gini',max_depth=3)
235 clf = clf.fit(X_train,y_train)
```

六、 分析與改進：

(1) 資料前處理方式：分析的預測表現會受到資料前處理很大的影響，適當的資料處理方式可以保留更多的原始資訊，使預測表現更好。

(2) 各類別樣本比例懸殊的問題：例如美國短毛貓種類就佔了貓的總數一半,所以我們針對此問題將短毛貓配上各自的顏色，讓此類別不會因為數量太大而影響預測的準確度。

(3) 參數調整：實作使用的的兩種預測模型產生的結果，Accuracy 都大約為 0.6，未來會再調整各種參數以增進我們的模型品質